Volume 07 Issue 02 Published, May 21, 2003 ISSN 1535-864X

Intel® Technology Journal

Intel® Centrino™ Mobile Technology

System-Level Validation of the Intel® Pentium® M Processor

A compiled version of all papers from this issue of the Intel Technology Journal can be found at:
http://developer.intel.com/technology/itj/index.htm

System-Level Validation of the Intel® Pentium® M Processor

Isic Silas, Mobile Platforms Group, Intel Corporation
Igor Frumkin, Mobile Platforms Group, Intel Corporation
Eilon Hazan, Mobile Platforms Group, Intel Corporation
Ehud Mor, Mobile Platforms Group, Intel Corporation
Genadiy Zobin, Mobile Platforms Group, Intel Corporation

Index words: Pentium® M processor, System Validation, Periodic SMI, Bus Marginality Testing

ABSTRACT

The Intel® Pentium® M processor is the first Intel microprocessor with major architectural and micro-architectural changes in performance and power and frequency optimizations, which directly entered the mobile market, i.e., without a previously validated desktop version. A very tight Intel Pentium M processor post-silicon period from first silicon to launch and lack of IA-32 testing and debug expertise by the Mobile OEMs, dictated a different approach to validate the CPU, at both the Intel level and the OEMs' level. The main goal was to ramp up the Pentium M validation capabilities very fast and uncover all silicon bugs before they were reported by OEMs (to prevent the time and effort it would take to debug sightings in the OEMs' environment), while maintaining the tight OEM development cycle from samples to launch.

This paper describes a novel complex post-silicon system validation methodology, that enabled us to deliver a healthy Intel Pentium M processor for the launch of the Intel® Centrino™ mobile technology, despite the tight validation schedule. This was achieved by Intel's uncovering all logic and circuit issues so no silicon-related bugs were found by customers, and by reducing dramatically the system-level-failures' debug time.

INTRODUCTION

The Intel Pentium M microprocessor, a key element of Intel Centrino mobile technology, is the first Intel processor with major architectural and micro-architectural changes in performance and power and frequency optimizations, which directly entered the mobile market, i.e., without a previously validated desktop version. Features of the Pentium M processor such as Enhanced Intel SpeedStep® technology, new branch prediction algorithms, and micro-ops fusion could potentially be a source of severe logic issues, while timing enhancements and low voltage working points (to reduce the CPU's power consumption) could result in marginality problems in the CPU's core and the I/O circuit.

The traditional CPU post-silicon platform-level validation cycle, at Intel, consists of parallel development and testing of system validation (SV), compatibility verification (CV), and OEMs programs.

The objectives of CV are to ensure that the microprocessor functions correctly in the standard system, with existing operational systems and commercial applications, and that it runs on customer reference boards equipped with commercially available (for the moment) hardware configurations. The drawbacks of using CV are insufficient visibility and controllability on CPU features usage and testing coverage, and difficulties in debugging of failures that in many cases are platform hardware or software rather than CPU-related.

The OEMs' primary goal is to maintain the tight customer development cycle from samples to launch. Their validation process is very similar to that of CV, with a focus on overall system functionality.

SV is based on testing the microprocessor functionality in specially designed, SV-hardware configurations, by running unique SV random tests that typically have nothing in common with any commercial software. The

special SV hardware aims to simulate all commercial platform configurations including all the add-in cards options. Similarly, the SV random testing aims to simulate all the existing, as well as yet to be developed, commercial software code combinations. These two factors provide very good controllability on validation coverage, since they allow for the accurate generation of any desired event and IA-32 scenario. The SV environment also assumes an efficient failure localization, which guarantees fast issue debugging and identification.

The very tight Intel Centrino mobile technology program schedule was mainly due to the short period between the Intel Pentium M processor first samples and launch. By not having debug hooks and expertise from the mobile OEMs, we had to approach the validation cycle in a different way at both the Intel and OEM level. The goal of SV was to ramp up the Pentium M processor validation capabilities in the fastest way and uncover all silicon logic and circuit marginality issues before they were reported by OEMs. Since OEMs got early Pentium M processor engineering samples (on average, approximately two weeks after SV got units) before validation was completed, uncovering issues before they were uncovered by the OEMs was a huge challenge. At the same time, an early enabling program was developed to allow OEMs to start their platform integration and validation; to test chipset, memory, and graphics subsystems; as well as other platform components, even before the Pentium M samples became available.

This paper describes different aspects of the Intel Pentium M processor's complex post-silicon validation program and details the major results.

THE INTEL PENTIUM M PROCESSOR SYSTEM VALIDATION

The Intel Pentium M processor system validation (SV) methodology is based on the following major components:

- A unique SV platform that differs from a standard motherboard because of specially developed hardware agents connected to the CPU front-side bus and other pins.
- A package of random instruction-based SV testing software, that ensures excellent coverage for logic and circuit marginality aspects.
- A Periodic SMI (PSMI) methodology for automatic and very fast reproduction of system-level failures on the Pentium M processor software model (RTL) or the Debug Tester.

Intel Pentium M Processor System Validation Platform

"Golan": Intel Pentium M processor/Intel® 855PM chipset SV Platform

The design of the Pentium M SV platform focuses on enhancing the following capabilities that are a must for qualitative and efficient validation of modern microprocessors:

- *Coverage/controllability*. This enables the most complex scenarios on CPU busses and other pins, to be easily programmed via testing software. Thus the processor bus behavior can be simulated, when used with various chipset types and add-in cards, by using only one type of chipset and without any add-in cards.
- *Determinism*. This ensures that you can immediately reproduce system failures, clock by clock.
- *Automation*. Automation allows for remote execution of SV tests and for remote/programmable modification (shmoo) of CPU operating parameters (e.g., voltage, frequency, temperature).

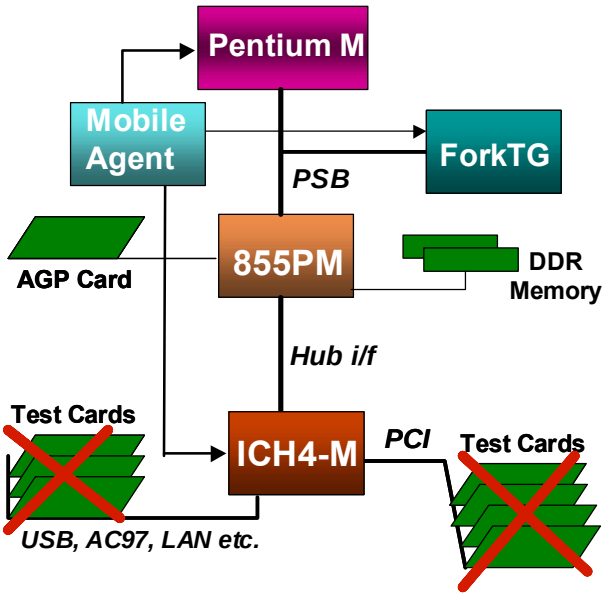

Figure 1: Block diagram of Golan, an Intel® Pentium® M processor/Intel® 855PM chipset system validation platform

A block-diagram of the Intel Pentium M processor configured with an Intel® 855PM chipset SV platform, named Golan, is shown in Figure 1.

The "brain" of this platform, the Mobile Agent, is implemented in a Field Programmable Gate Array (FPGA) and controls a wide set of features to support all SV requirements. The essential part of the Mobile Agent is the Programmable Signal Generator (PSG). This device

was designed to enable simultaneous injection of signals during CPU code execution. Those signals, which are activated in different modes (e.g., single or burst, level or pulse), include legacy signals like NMI, SMI, and INTR, which are connected directly to the processor pins, and various break events, which are sent from the IO Control Hub (ICH). Each PSG signal duration and timing parameter can be programmed via software. In addition, the Mobile Agent can identify different power management events and use them as trigger sources for PSG signals. It guarantees good coverage of various border scenarios, essential for mobile platforms, that previously were subject to real CPU problems or other platform component issues. It should be emphasized that PSG completely eliminates the need for commercial Peripheral Component Interconnect (PCI) or graphics cards for the SV board, ensuring at least the same level of CPU bus stressing and significantly better reproducibility.

The Golan SV board also includes a power management Rule Checker/Tracker designed to monitor the CPU's and the system's power management modes transitions, and the Enhanced Intel SpeedStep rules.

Very accurate, automated control over the system clock scheme enables the CPU Processor System Bus (PSB) frequency modification (shmoo), which is widely used by SV for Pentium M circuit marginality testing, together with the Core voltage control (shmoo) capability, which is also implemented in the Golan board.

Among other Golan SV board features, are power consumption (V,I) measurements and history recording, reset and straps, and a Joint Test Action Group (JTAG) standard controller.

Fork Transactions Generator: SV Processor System Bus (PSB) Agent

Another leading methodology for the Pentium M processor SV platform is the usage of active bus agents to provide bus activation right on the spot. The biggest advantage of these agents over end point agents (such as PCI cards) is their capability to inject aimed transactions, triggered by normal bus events, and their high success rate in failure reproduction. The most important active bus agent used for the Pentium M processor validation was the Fork Transactions Generator (Fork TG), residing on the processor system bus (PSB).

Because it was designed for mobile systems, the Pentium M processor bus supports uni-processor mode only, thus, preventing the usage of another CPU for its caches and bus unit stressing. Therefore, Fork TG, implemented as a re-programmable FPGA, demonstrates a new concept in the architecture of a uni-processor validation platform. Specially designed to be placed between the processor and the Memory Control Hub (MCH), Fork TG is acting in a true uni-processor environment. Nevertheless, neither the processor nor the MCH are aware, logically, of its presence. When injecting PSB transactions, Fork TG manipulates each of the two chips to act as if the counter-chip is the source for those transactions. This innovative solution brings plenty of new features and benefits to the system validation environment.

Fork TG activity is based on controllable triggering and inter-operating injection streams. The notion of injection implies placing transactions on the processor system-bus and thus creating traffic. The density of this traffic as well as its duration are regulated by Fork TG configuration parameters programmed via validation software as part of the SV tests.

Three modes have been defined in order to regulate Fork TG injection streams: pre-defined, shadow, and self-generated. The pre-defined mode is mainly used to exercise the cache cluster (L1 and L2) and the Memory Ordering Buffer (MOB) in the processor. In this mode, Fork TG monitors the bus, looking for an event (e.g., a bus transaction) that triggers it to stream the pre-defined package of transactions, which were programmed beforehand to Fork TG's external RAM.

The shadow mode was conceived for exercising the cache cluster of the CPU, along with the processor bus unit. In this mode the transaction sampled on the PSB undergoes some updates in Fork TG, and the newly created transaction is placed back on the bus as a part of the Fork TG injection stream. In other words, the original transaction is 'shadowed' by the newly generated one, altered according to some rule. This is the essence of this operation. Possible alteration rules, chosen during SV test generation, include replacement of the request type (e.g., a Read Invalidate operation instead of a Read Data operation), replacement of the memory type, a change in the length of the request, or an update of the transaction address within the page (4K) boundary.

In Self-Generated mode, the PSB is stressed by bursting out injections irrespective of current bus events. Three parameters, start address, end address, and address step, along with the number of loops, set the duration of the self-generated test.

The Fork TG has been heavily used throughout the Pentium M processor post-silicon validation period, mostly for caches and the PSB stressing. This resulted in the detection of two real silicon issues.

Random Instruction Testing: Pentium M processor System Validation Methodology

There are several different methodologies for processor system-level validation. The Pentium M processor system validation was principally based on the Random

Instruction Testing (RIT) concept. This means that CPU correct functionality, logic and circuit wise, is being checked by a tremendous amount of discrete tests aimed at covering all possible architectural and micro-architectural scenarios enabled by Spec. Each random test (seed), typically runs from reset, emulates some artificial operational system with all its attributes (e.g., descriptor and interrupt tables, exception handlers, paging hierarchy), and also contains several thousand instructions of "user's code," running under this pseudo-Operating System (OS). Both OS attributes and user's code are maximally random, in terms of instruction sets, memory allocation for OS level structures, code and data, caching policy, etc. Random programming of SV platform residing agents, namely, Mobile Agents (signals, events) and Fork TG (processor system bus transactions) are also under seed responsibility.

Random test generation is being done in a software environment, specially developed at Intel. It includes a kernel, which is applicable to any current and future microprocessor, and modules that define processor-specific test content. These modules, which, by themselves, are very complex software projects, are responsible for the coverage of microprocessor health. Test generation rules implemented in these modules result in the success of silicon in mass production, e.g., absence of escaped bugs.

The Pentium M SV test content has been developed, taking into account these processor multiple architectural and micro-architectural enhancements, as compared to previous generations of CPUs, and the unique mobile features of the Pentium M processor. SV engineers spent much time with architects and designers in order to identify the most risky areas of this processor, before implementing their coverage in testing software. Among the areas that required special attention were the following:

- Enhanced Intel SpeedStep technology assuming processor frequency transitions during regular code execution. CPU response to different events, like power management or external interrupts randomly occurring during any of the transitions, was a special focus of SV
- Various scenarios related to power management transitions of the CPU, e.g., to/from sleep and deep sleep states, with heavy involvement in break events, interrupts, APIC messages, and snooping
- Novel micro-architectural features, e.g., micro-ops fusion
- Marginal data paths capable of causing system failures, specifically at high frequencies

Random Instruction Testing methodology assumes the execution of many millions of seeds before the required coverage of all CPU features is achieved. Therefore, the SV Lab infrastructure includes up to 100 Golan platforms and several hundred test generators able to produce seeds 24-hours a day, executing approximately 1 trillion instructions per week. Enormous effort was put into enhancing generation and execution throughput, resolving Lab network problems and achieving automatic control over lab resources. As a result, around 10 billion random seeds, each containing several thousand random instructions, were run by the Pentium M processor throughout its validation period from first silicon arrival to launch.

One more important thing that influences the post-silicon validation success, especially during the first critical weeks, is the cleanliness of testing software. To ensure the software is clean, the Pentium M processor test content has passed massive dry runs on Intel Pentium® 4 platforms, on Golan SV boards equipped with the Erez interposer, described below, and on Pentium M models. As a result, when silicon arrived, the system validation team faced few false alarms and could concentrate just on debugging real CPU issues.

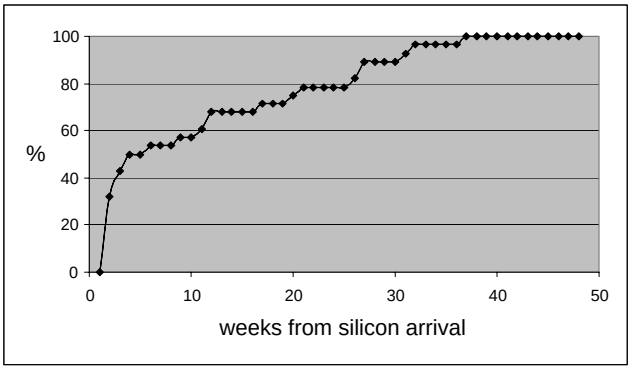

Figure 2: Cumulative bug detection rate by Intel Pentium® M processor SV from first silicon to launch

All factors described in this section have resulted in very fast bug rate detection once Pentium M processor samples arrived at the SV Lab. Figure 2 shows the cumulative bug detection rate from first silicon to launch, and it clearly shows that 50% of the logic issues found in the Pentium M processor were uncovered during the first two to three weeks of validation. Also, the latest bug that needed a silicon fix was uncovered after 20 weeks of validation, while all further issues were of low severity and were closed via errata.

Periodic SMI (PSMI) Failures Reproduction Methodology

An essential component of any silicon validation program is the capability to quickly and efficiently reproduce system failures on the RTL model of the chip under test, in the case of a logic issue, or on the Tester, in the case of a circuit issue. The major problem of reproduction is to ensure that silicon behavior on the system is absolutely synchronized with that of the RTL/tester. This is because most failures occur far beyond the reset point, while all processor arrays and registers, e.g., caches, TLBs, branch predictor, contain some history, which is not visible externally. The Periodic System Management Interrupt (PSMI) methodology, developed at Intel approximately five years ago, is aimed at resolving this problem and providing full synchronization between microprocessor behavior on the system and simulation environment during the failure reproduction phase. This methodology assumes an injection of periodic signals on the SMI pin of the CPU, while the latter executes a failing test or application, forcing it to enter periodically into a specially developed SMI handler, as shown in Figure 3.

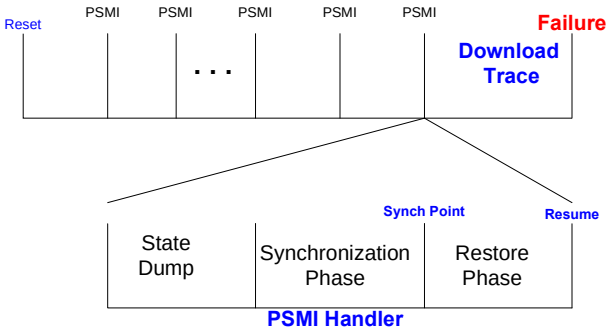

Figure 3: Schematic description of periodic SMI methodology

The Logic Analyzer trace, downloaded for further reproduction, should reflect full CPU external activity (taken from all its pins) starting from the SMI closest to the failing point. The most critical parts of the sophisticated PSMI handler are the Dump and Synchronization Phases as shown in Figure 3. They provide a dump of the content of the CPU internal state to memory, making it visible on the processor system bus, and an initialization/flush of some arrays and state machines. Therefore, the CPU behavior, after the Sync point of the handler, is absolutely deterministic, and the trace is ready for conversion and reproduction on the RTL model or Tester.

PSMI technology has been successfully used for system-level validation of Intel Pentium® MMX™, Pentium® III and Pentium® 4 processors. The mobile-oriented features of the Intel Pentium M processor brought new challenges derived from the following facts:

- SMI is heavily used in mobile platforms; therefore, the potential conflict between regular SMI and PSMI should be eliminated. This has been done by separating SMI and PSMI both in the internal CPU design, and at the system memory space level.
- Enhanced Intel SpeedStep technology, implemented in the Pentium M processor assumes internal clock frequency changes during regular CPU running. These frequency transitions have to be captured by the PSMI handler and transferred correctly to the simulation environment.
- Power management features that cause partial shutdown of Pentium M processor internal and external clocks are heavily used in the CPUs of mobile platforms. These also introduce serious reproduction issues.

In order to ensure full readiness of the PSMI flow in all corner cases, and to accommodate the above challenges, PSMI was validated extensively in the pre-silicon period. A comprehensive test plan was written, which included 60 focus tests and over 100 random tests. Each test emulated the PSMI flow–a trace being taken from a simulated system and driven into the target simulation environment. A simulation acceleration machine was heavily used to increase the execution throughput.

Despite the above increased complexity, the PSMI methodology has demonstrated impressive results during Pentium M processor system-level validation. The PSMI flow was ready for use and reproduced the first logic issue the first week after first silicon arrival. Overall, the post-silicon validation period, 100% (!) of all the logic issues brought from the system, have reproduced on the RTL model, with an average latency of only 1-2 days. As for circuit issues mostly related to internal speed paths, Intel's success rate is approaching 94% – much higher than in any previously validated microprocessors.

PROCESSOR SYSTEM BUS MARGINALITY VALIDATION

Due to the low-voltage and high-frequency Processor System Bus (PSB) of the Pentium M processor, a thorough signal integrity validation became critical to ensure the design robustness of customer reference boards and OEM's mobile systems. Typically, bus marginality is measured in two domains, namely, input buffer reference

voltage Vref, and data/strobe timing Δt. A two-dimensional (Vref, Δt) area of reliable system functionality forms an ellipse, called an Eye Diagram, as shown in Figure 4.

One can see that the boundaries of an Eye Diagram reflect conditions where the system starts failing, while its center provides an optimal design point for the Vref and Δt parameters. A single Eye Diagram measurement requires several hundred iterations to be accomplished at different points, while measurements should be repeated under a wide spectrum of conditions and interconnection parameters, e.g., temperature, CPU or chipset skew units. This makes the bus marginality testing extremely time consuming (typically, about four to eight hours per "eye").

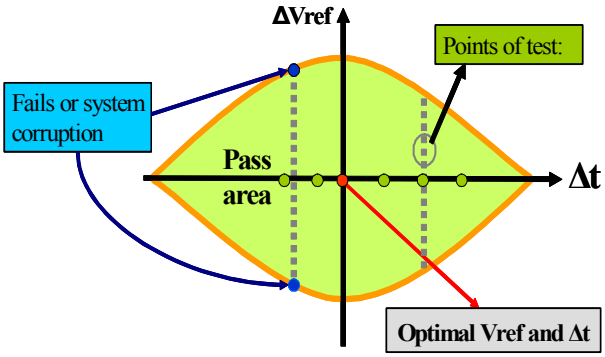

Figure 4: An example of a bus marginality "Eye diagram"

To intensify the process of Pentium M processor analog validation, a novel tool, named the Bus Marginality Tester (BMT) was developed by the Pentium M Signal Integrity and Analog Validation team. The BMT is actually a package of hardware and software tools that is used to measure automatic signal integrity bus marginality. Its structure is hierarchical, as shown in Figure 5, consisting of three levels: Target, Host, and Server. Bus stress runs on the Target level (actually a board under test), Host controls the Eye Diagram building process and is connected to the Target via a PCI-to-PCI bridge. It regulates Eye Diagram parameters on the Target level via a specially designed margin card (Vref) and BIOS settings (Δt). A single server manages several Target/Host blocks via a Local Area Network (LAN) and provides the user interface with an operator. It allows massive automatic parallel execution on several Target/Host blocks from one server. All Eye Diagram accumulated information coming to the server is stored in a special database and is subject to automatic sophisticated analysis.

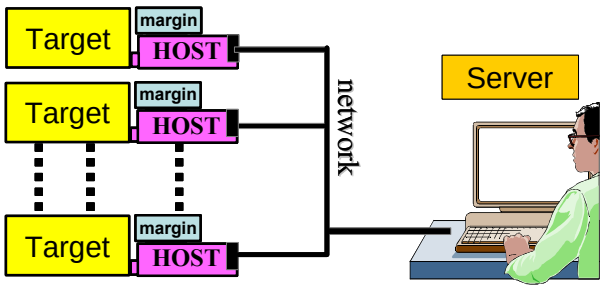

Figure 5: Block diagram of Bus Marginality Tester (BMT)

BMT usage in the Pentium M processor project brought very impressive results. First of all, an average Eye Diagram measurement time was reduced from hours to ten to fifteen minutes. It allowed us to build about 40,000 functional "eyes," providing wide coverage of bus interconnect validation, with a Shmoo of many parameters: package and board impedance, silicon skew material, temperature, IO voltage, various bus stress patterns, various memory combinations, etc.

BMT has uncovered and helped to resolve several serious system problems. Among them are two severe Small Outline Dual In-line Memory Module (SO-DIMM) memory issues, Double Data Rate (DDR) Logic Analyzer Interface issues, and DDR data/strobe de-centering problems. The BMT was effectively used for the fast comparison of signal integrity bus robustness design of different OEM motherboards. Thus, we were able to give OEMs early feedback on the robustness of their board design.

Erez Interposer: Early Enabling Program

A very tight Intel Pentium M processor post-silicon period from first silicon to launch has put significant pressure on OEMs' platform testing and integration schedule. A major breakthrough in this area was early availability of a specially designed small form-factor, full speed, pin compatible interposer, named Erez. Equipped with a Pentium 4 processor, the Erez interposer was inserted into the CPU socket of the Pentium M system, and it successfully imitated the functionality. This program paved the way for platform designers of OEM systems, reference boards, and SV platforms to debug their hardware and ramp-up the critical mass of systems, months before Pentium M processor silicon arrived. In total, eight OEMs have participated in the early enabling program, using Erez interposers. Each OEM used it for the development of several mobile platforms. In parallel, it also enabled BIOS and software developers to test and clean their products.

RESULTS SUMMARY

An efficient System Validation program has been developed for the Intel Pentium M processor, a key element of Intel Centrino mobile technology. The objective of system validation (SV) is to ensure the delivery of a healthy Pentium M processor, from the point of view of logic and circuit marginality, in an extremely tight schedule, taking into account that it is the first time a brand new processor enters the mobile market, without any sophisticated IA-32 testing and debug capabilities.

Intel uncovered 100% of the Pentium M logic and circuit issues with no silicon-related bugs found by customers. In addition, 50% of the tens of the silicon issues were detected during the first two weeks of validation, showing the high quality of the SV test suite and the excellent throughput of moving failures from system to RTL and Tester, using the PSMI tool. A combination of a unique SV test environment, the PSMI tool, Erez early enabling program, and the BMT allowed Intel to deliver the Pentium M processor in time for the launch of Intel Centrino mobile technology.

DISCUSSION

The success of the Intel Pentium M processor system validation program was predetermined by the following factors:

- Validation engineers had extensive expertise in processor and platform-level architecture, processor testing, and validation platform design
- Validation platforms and testing software were defined and developed in a very cooperative atmosphere
- Innovative coverage and validation efficiency-oriented ideas (Fork Transactions Generator, Bus Marginality Tester, and Erez) were driven and implemented
- Very thorough preparations were made to guarantee the readiness of the validation platform, testing software, and debugging tools (PSMI) on the eve of silicon arrival

As modern microprocessors become more and more sophisticated, and as market competition tightens the post-silicon schedule, it is essential to combine efficiently all the above factors, as was done for the Intel Pentium M processor.

AUTHORS' BIOGRAPHIES

Isic Silas received his B.Sc. degree in Electrical Engineering from the Technion (Israel Institute of Technology) in 1988. He is the Intel Israel Mobile Group, System Assurance department manager, responsible for developing hardware and software capabilities for components and platform-level testing, to ensure the delivery of healthy products and platforms. Isic joined Intel in 1988, working on many aspects of platform validation. He started as a hardware engineer developing validation platforms, and then moved on to validation software and validation and debug tools. He is now perceived as an Intel expert in the area of validation. His e-mail is isic.silas@intel.com

Igor Frumkin is a validation group manager in Intel's Israel Design Center, currently leading system-level validation of Intel microprocessors for mobile platforms. Igor joined Intel in 1995. He received his B.Sc. and M.Sc. degrees in Electrical Engineering from the Institute of Aviation Instruments Making in Leningrad (the former Soviet Union) in 1981, and his D.Sc. degree in Quality Assurance from the Technion (Israel Institute of Technology) in 1995. His e-mail is igor.frumkin@intel.com

Eilon Hazan is a debug tools team manager in Intel's Israel Design Center, leading the PSMI tool and other debug capabilities for system-level validation in the Mobile Platform Group. Eilon joined Intel in 1995. He received his B.Sc. degree in Electrical Engineering from the Technion (Israel Institute of Technology) in 1997. His e-mail is eilon.hazan@intel.com

Ehud Mor is a Hardware Development group manager in Intel's Israel Design Center, responsible for the design of hardware solutions for system validation. Mor received his B.Sc. degree in Electrical Engineering from the Technion (Israel Institute of Technology) in 1979. Before joining Intel in 1994, Mor was a senior design engineer at Integrated Device Technology Inc., RAFAEL and Elbit Electronics. His e-mail is **ehud.mor@intel.com**

Genadiy Zobin is the Signal Integrity Engineering Group manager in the Intel Israel Development Center. He works on interconnect signal integrity design and validation of mobile platforms. Genadiy joined Intel in 1991. He has wide experience in platform design and validation, including logic, circuit, ASIC, and interconnect design. He received his M.Sc. degree in Electrical Engineering from Leningrad Polytechnic Institute in 1977. His e-mail is genadiy.zobin@intel.com

For further information visit:

developer.intel.com/technology/itj/index.htm